LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

LLNL-TR-664515

Automatic Complexity Reduction

D. A. White, M. Stowell, K. Lange

November 19, 2014

Disclaimer

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

Automatic Complexity Reduction

Daniel White

November, 2014

1. Abstract

This is the final report for LDRD project 12-ERD-038 titled Automatic Complexity Reduction. The original budget request was for $1.1M for three years, the project was instead funded at $420K for three years. The team consisted the PI Daniel White, post-doctoral researcher Kyle Lange, and staff member Mark Stowell. This work largely builds upon the research of Matt Stephanson, who was a summer scholar at LLNL. This report will provide a high level overview of the problem and the solution approach, as well as results, and future research directions.

2. Approach

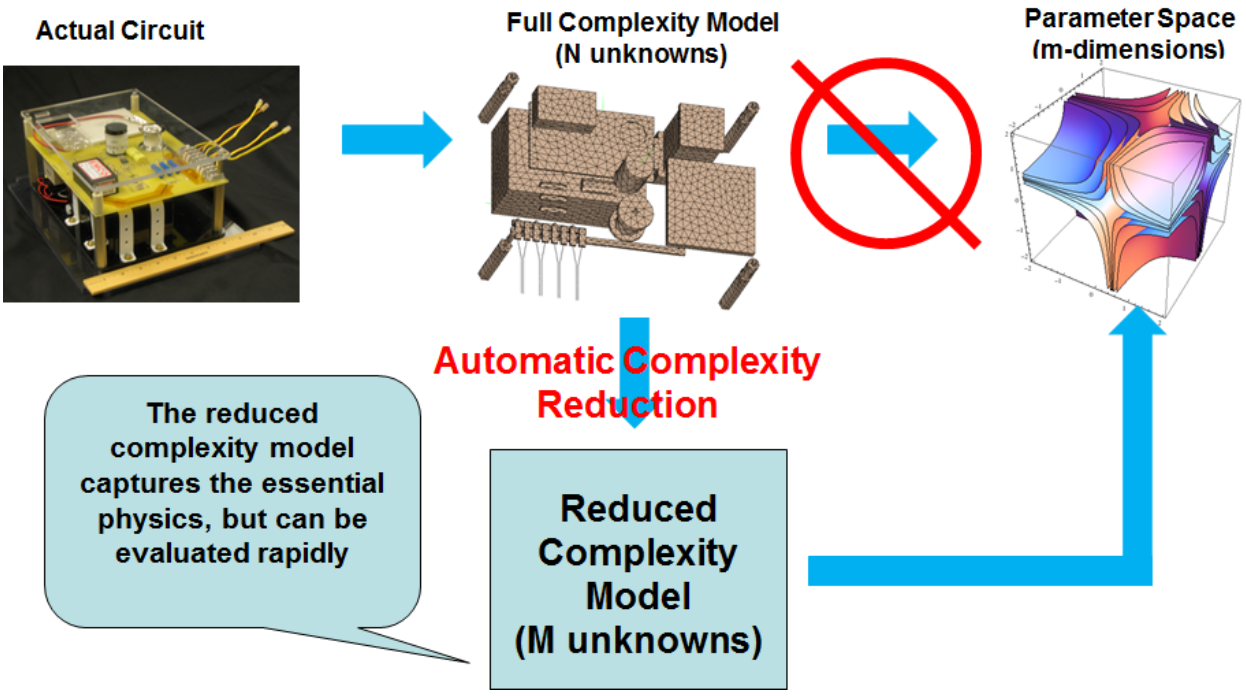

Figure 1. Overview of the model order reduction problem.

The generic problem to be solved is illustrated in Figure 1 above. On the far left we have some system (could be a complicated assembly, or a single waveguide or antenna) and we want to know how this system responds to an electromagnetic input. The system is parameterized by a few parameters, the parameters could be frequency and angle of incidence of the excitation,

could be geometric dimensions, material properties, etc. The full complexity model is a finite element or boundary element discretization, and for purposes considered here the solution of the full complexity model is the "truth" or "exact" solution. The goal is to obtain solutions for arbitrary combinations of parameters, perhaps thousands or millions of points in parameter space. For example, if we have 3 parameters, and each parameter can take on 100 values, this gives 1 million required simulations. This is problematic if each simulation requires minutes to hours. And clearly the problem becomes quite complex as the number of parameters increases; the computation cost (complexity) is an exponential function of the number of parameters. This is sometimes referred to as the curse of dimensionality.

The proposed approach is also shown Figure 1. Instead of evaluating the full complexity model at every point in parameter space (the top row in the figure), the full complexity model is evaluated at a few key points in parameter space called sample points, the resulting solution at each sample is called a snapshot. The snapshots are used to construct a reduced dimensional model, also called a reduced order model. The idea is that this reduced order model captures the essence of the full complexity model but is orders of magnitude more efficient to evaluate. It is important to note that this is not a simple response surface, the reduced order model provides the full dimensional solution. For example, if the full complexity model produces the vector field for every element in the mesh, then the reduced order model will also produce this same field. Some of the key questions are, how many shapshots are required to create the reduced order model, how accurate is the reduced order model, and which particular interpolation methods should be used.

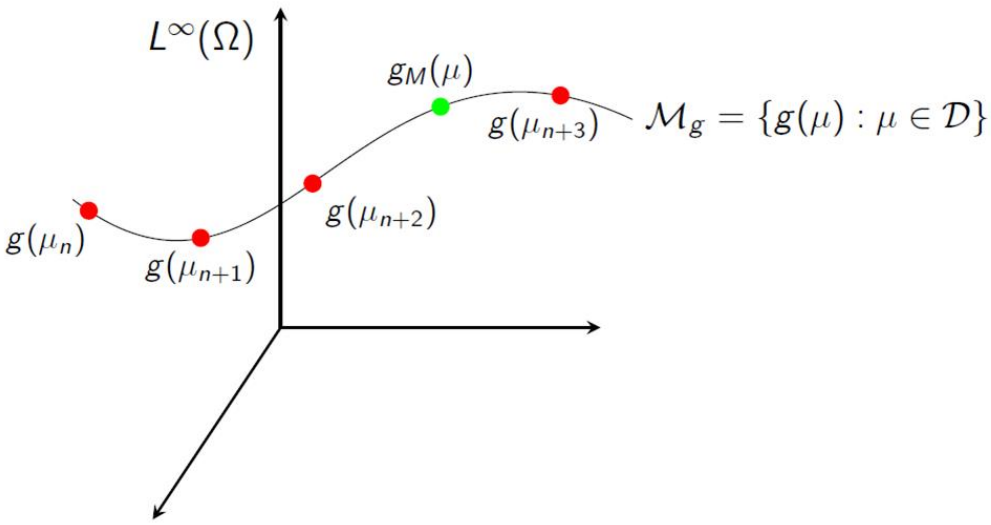

Figure 2. Illustration of a reduced dimensional manifold.

The first step in automatic model order reduction is to perform a change of basis. The original basis is the finite element or boundary element representation. The exact solution is a point in

R^N, where N is the number of unkowns (degrees of freedom). The key assumption is that as the parameters are varied the exact solution resides in a reduced dimensional manifold in R^N, i.e. a one parameter problem results in a curve in R^N, a two parameter problem produces a surface in R^N, etc. Therefore some efficiency can be gained if we only approximate the reduced dimensional manifold instead of all of R^N. We seek a change of basis that will approximate this reduced dimensional manifold. One change of basis is a truncated proper orthogonal decomposition using the above mentioned snapshots.

Let the electromagnetic problem, after discretization by an appropriate finite element or boundary element method, be given by

$$A(p)x = b(p) \quad p = \{p_1, p_2, \ldots, p_m\}$$

where $p = p_1, p_2, \ldots, p_m$ are the parameters, $A(p)$ is a *NxN* matrix, $b(p)$ is the right hand side vector of length *N* representing source terms and boundary conditions, and *x* is the solution vector of length *N*. Note that we need to solve over and over again, K^m times, where *m* is the number of parameters and *K* is the number of samples per parameter.

The change of basis is given by

$$x = Vx'$$
$$V^T A(p) V x' = V^T b(p)$$

where V is a NxM matrix, with M << N. Therefore the matrix V^T A V is of dimension MxM, much smaller than the original NxN matrix A. The columns of V are the above mentioned snapshots, solutions of the full complexity solution at selected sample points. If it is desired to have the smallest possible M, the singular value decomposition can be used to keep the minimum number of columns of V to obtain a prescribed tolerance. When the SVD is used, this is called a truncated proper orthogonal decomposition.

Note that the matrix triple product V^T A(p) V is expensive to compute. We do not want to explicitly compute this product for arbitrary values of p. If there exists a compact affine approximation for A(p), then this approximation can be used. As a simple example, consider

A(p) = A0 + A1 p + A2 p^2.

where A0, A1, and A2 are constant matrices independent of p. Inserting this into the change of basis formula gives

V^T A(p) V = V^T A0 V + V^T A1 V p + V^T A2 V p^2.

Clearly the matrix triple products are independent of p and can be compute just once. Given an arbitrary value of p, the above MxM matrix is formed and solved, and the full solution is then given by x = Vx'. This assumption of affine approximation is not academic, certain problems in elasticity, heat transfer, electrostatics have exact affine representations of the finite element stiffness matrix when the parameter p denotes either a material property or a geometric dimension. In this project, we are concerned with problems that do not have an exact affine representation, and we therefore seek approximations. Two such approximations are the radial basis function approximation, and the empirical interpolation method.

We investigate radial basis function interpolation as a means to construct an approximate affine representation of A(p). Radial Basis Functions (RBF) are smooth functions of a r = |x - xp|, where xp is a given point. RBF's are not explicitly interpolatory, but by solving a system of equations they can be used to fit any smooth function. The advantages of RBF's are that they are easily extended to high dimensional spaces, and they exhibit exponential convergence as more functions are used. Common RBF's are shown below.

Gaussian	$e^{-(r/a)^2}$
Inverse Quadratic	$\left(1+(r/a)^2\right)^{-1}$
Multiquadric	$\sqrt{1+(r/a)^2}$
Inverse Multiquadric	$\left(1+(r/a)^2\right)^{-1/2}$
Thin Plate Spline	$(r/a)^2 \log(r/a)$
Polyharmonic Spline	$(r/a)^{2p} \log(r/a)$

Given an arbitrary function of p, the radial basis function approximation of f(p) is given by

$$f(\mathbf{p}) \approx \tilde{f}(\mathbf{p}) = \sum_{i=1}^{M} \alpha_i \varphi(\|\mathbf{p}-\mathbf{p}_i\|) + \sum_{j=1}^{K} \beta_j \psi_j(\mathbf{p})$$

where ϕ is the radial basis function, pi is the ith interpolation point, and αi is the coefficient associated with basis function i. It is common to add a few low order polynomials into the interpolation, these are given by Ψ. Given f(pi) for i=1,m, the coefficients are given by the system of equations below.

$$\begin{pmatrix} \mathbf{R} & \mathbf{P} \\ \mathbf{P}^T & 0 \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} \mathbf{f} \\ 0 \end{pmatrix}$$

$$\alpha = (\alpha_1, \alpha_2, \ldots, \alpha_M)$$

$$\beta = (\beta_1, \beta_2, \ldots, \beta_K)$$

$$\mathbf{f} = (f(\mathbf{p}_1), f(\mathbf{p}_2), \ldots, f(\mathbf{p}_M))$$

$$R_{ij} = \varphi(\|\mathbf{p}_i - \mathbf{p}_j\|) \text{ for } 1 \le i, j \le M$$

$$P_{ij} = \psi_j(\mathbf{p}_i) \text{ for } 1 \le i \le M,\ 1 \le j \le K$$

Note that this system of equations is only computed once for the unknown coefficients α and β. This then defines the approximate affine representation of A(p), and the reduced order model is completely defined. The offline steps are 1) compute the full solution at a few sample points, 2) compute the RBF approximation of the matrix A(p), 3) compute the reduced order model. Then for every parameter p the online steps are 1) form the reduced order model, 2) solve the model, 3) construct the approximate full solution. This algorithm was implemented in software and results are shown below. One issue that was discovered was that the system of RBF equations can be ill-conditioned, leading to a poor interpolation. The solution was to limit the size of the RBF interpolation by subdividing the parameter space. In simple terms, the entire parameter space is spanned by several reduced order models rather than one large model. A hierarchical sampling structure was used to subdivide the parameter space. We believe this work is novel and it has been submitted for publication.

The above described RBF interpolation is appealing because it is a "black box" method, it does not require any modifications to the finite element or boundary element solver, only matrices and solution vectors are required. But there are other methods that may yield better results if additional information is made available. In particular we investigated empirical interpolation, also known as magic points interpolation. This is a greedy algorithm that constructs an optimal interpolation given any initial basis set. It is a well conditioned and stable algorithm. The outline is given below.

The points zi are the magic points. These points are not known in advance, they are compute one at a time by minimizing the L_infinity norm. In our case, the magic points zi are matrix locations, and f(zi) are matrix entries. This means that when we wish to solve a parameterized problem, we don't need to compute the full finite element or boundary element matrix, we only need to compute a select set of matrix elements, and from these we can reconstruct the entire

solution. The functions ζ are the finite element or boundary element matrices (which can be thought of as being reshaped into vectors), and the q are ideal basis for these matrices.

Given set of functions $\{\xi_i\}_{i=1}^m$ find new points $\mathbf{z} = [z_1,\ldots,z_m]^T$ and new basis $\{q_i\}_{i=1}^m$

Definitions

$$\mathbf{z}^L = [z_1,\ldots,z_L]^T$$
$$\xi_L(\mathbf{z}^{L-1}) = [\xi_L(z_1),\ldots,\xi_L(z_{L-1})]^T$$
$$\mathbf{Q}^{L-1}(x) = [q_1(x),\ldots,q_{L-1}(x)]$$
$$(\mathbf{Q}^{L-1}(\mathbf{z}^{L-1}))_{ij} = q_j(z_i)$$

set $z_1 = \arg\ \sup_{x\in\Omega}|\xi_1(x)|$

set $q_1(x) = \dfrac{\xi_1(x)}{\xi_1(z_1)}$

$$\mathbf{Q}^1(\mathbf{z}^1) = q_1(z_1) = 1$$

for $L = 2,\ldots m$

solve $\mathbf{Q}^{L-1}(\mathbf{z}^{L-1})\rho^{L-1} = \xi_L(\mathbf{z}^{L-1})$ for ρ

define $r_L(x) = \xi_L(x) - \mathbf{Q}^{L-1}(x)\rho^{L-1}$

set $z_L = \arg\sup_{x\in\Omega}|r_L(x)|$

set $q_L(x) = \dfrac{r_L(x)}{r_L(z_L)}$

Given $\mathbf{z} = [z_1,\ldots,z_m]^T$ and $\mathbf{Q}^m(x) = [q_1(x),\ldots,q_m(x)]$

we can interpolate

$$f(x) \cong \hat{f}_m(x) = \mathbf{Q}(x)\mathbf{Q}(\mathbf{z})^{-1} f_m(\mathbf{z})$$

Note that by construction Q is a triangular matrix with 1's on the diagonal, hence the solve is efficient and stable. Perhaps not obvious, the above interpolation results in

A(p) = A0 f(z0) + A1 f(z1) + ... Am f(zm)

where the Ai's are constant matrices independent of the parameters p, and the f(zm)'s are particular values of the FEM/BEM matrices that are of course functions of p. Note that the function form of the p dependence is not needed. Note that this is an invasive method, the FEM/BEM code must be modified to be able to compute only selected values of the FEM/BEM matrix for a given parameter. This is easy to do with our in house codes EMSolve and EIGER, but not possible with commercial codes.

3. Circuit in a Box Results

Consider the following scenario. We know that a particular electronic circuit can be negatively affected by an incident EM wave, we know this experimentally. The frequency is very important, let's assume the effect is strongest at 500MHZ. The measurements are made in a anechoic chamber, with the circuit sitting on Styrofoam. The important factor is the current induced on a particular microstrip. The complication is that in the real world circuit boards don't float in space, they are typically enclosed in imperfect chassis, and the details of the chassis may not be known. As an example problem, a specific circuit board (a 10MHz clock circuit, common to many industrial controls) is placed in a PEC rectangular box with a circular aperture. The location of the circuit (X,Y, and Z) as well as the location of the aperture are parameters. A frequency domain boundary element method is used by the LLNL EIGER code to solve this problem. The total number of mesh elements is 3320, the total number of unknowns is 4122. This may not seem large, but the solver scales as O(N^3), so a single solve requires several minutes.

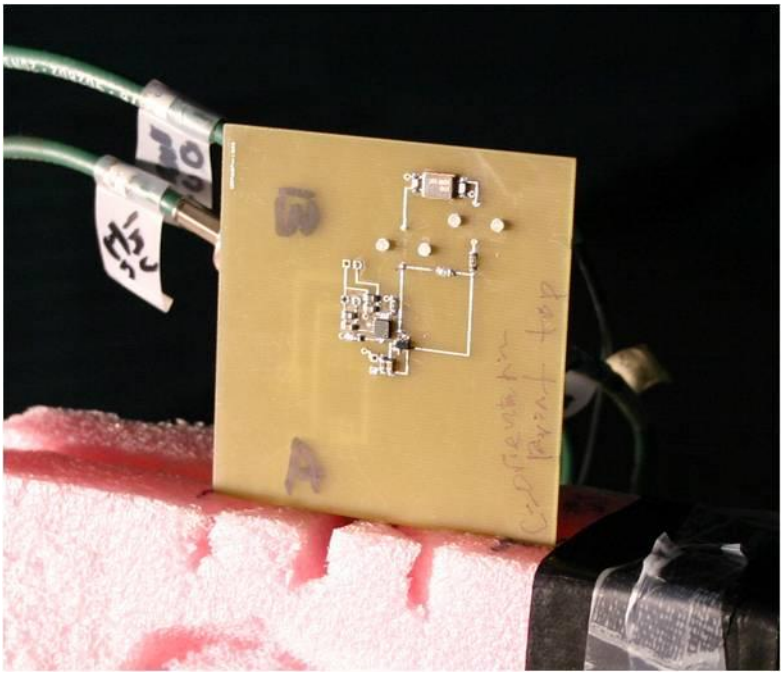

Figure 3. a 10MHz oscillator circuit being tested in an anechoic chamber.

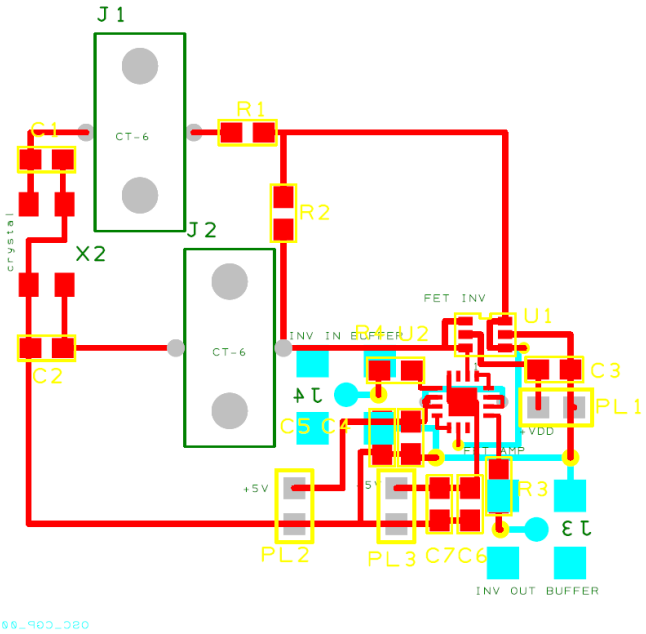

Figure 4. Layout of the 10MHz oscillator circuit.

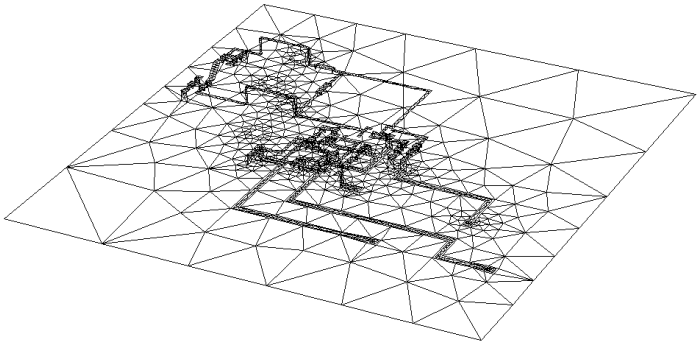

Figure 5. Boundary element mesh of the 10MHZ circuit.

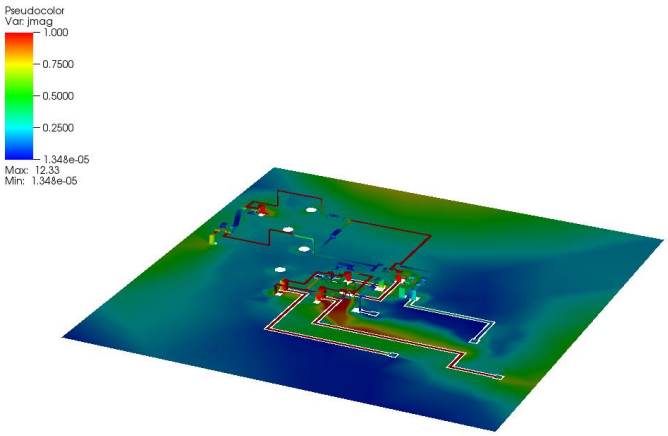

Figure 6. Computed current on the 10MHz circuit at 500MHZ.

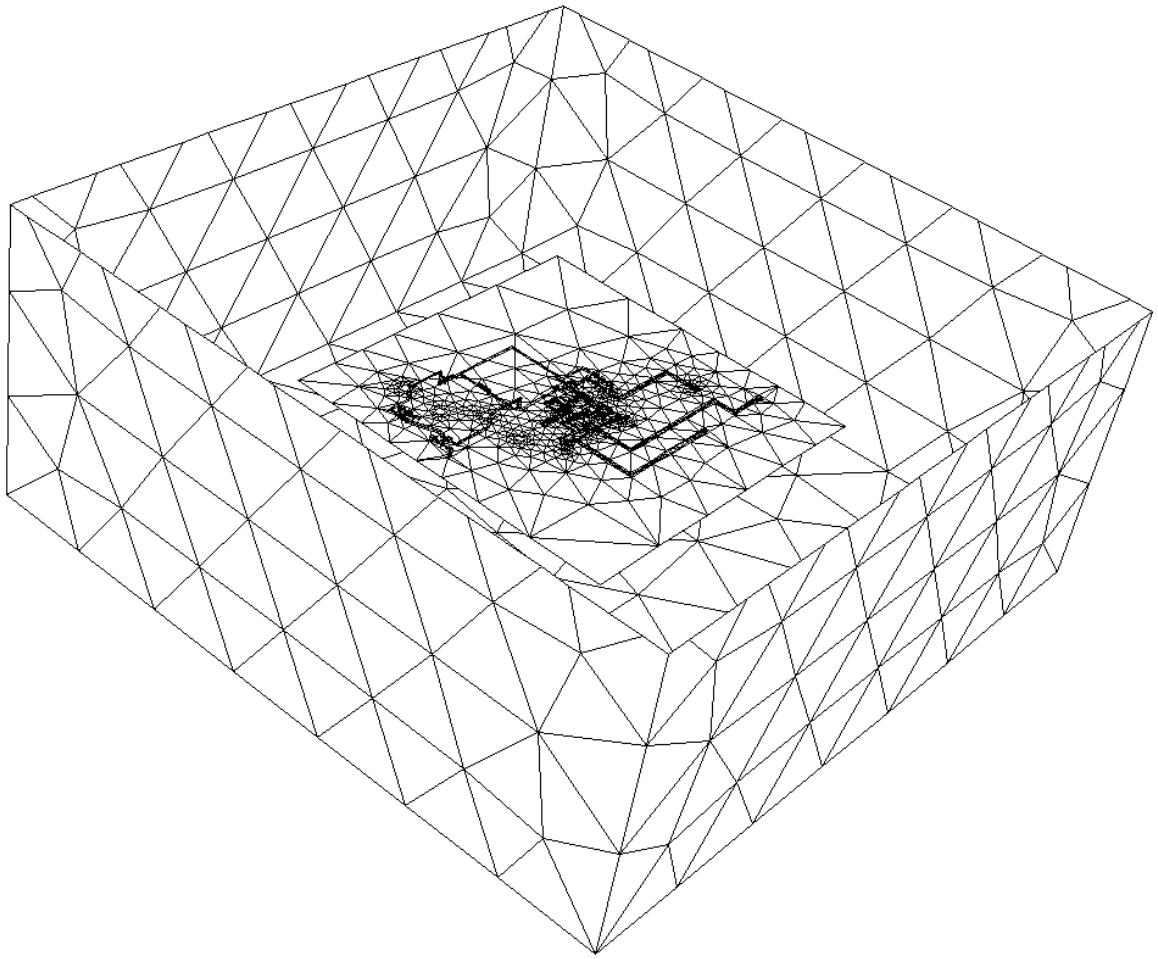

Figure 7. Computational mesh of the circuit in a box. The (X,Y,Z) position of the circuit is a parameter to be varied.

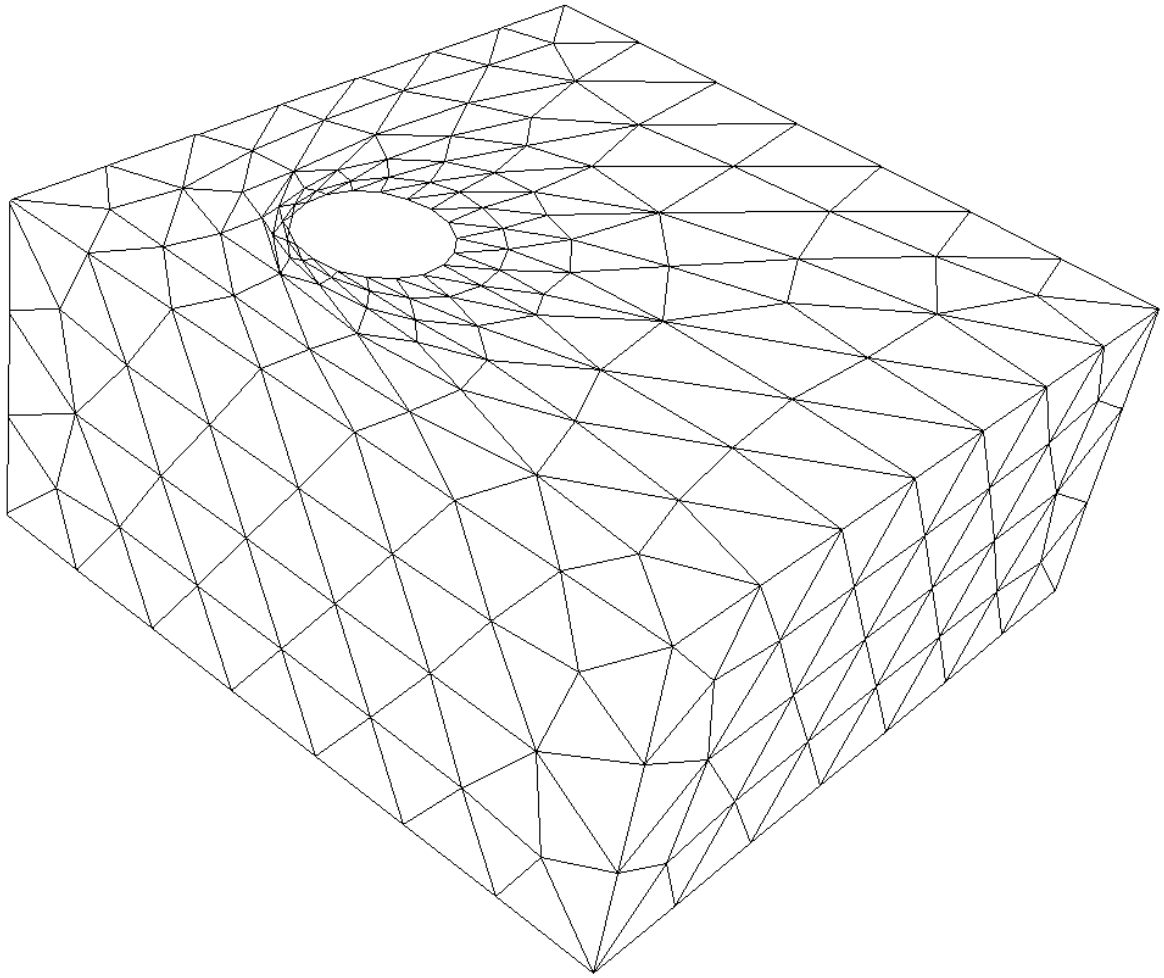

Figure 8. The top of the box showing the aperture. The (X,Y) position of the aperture is a paramter to be varied.

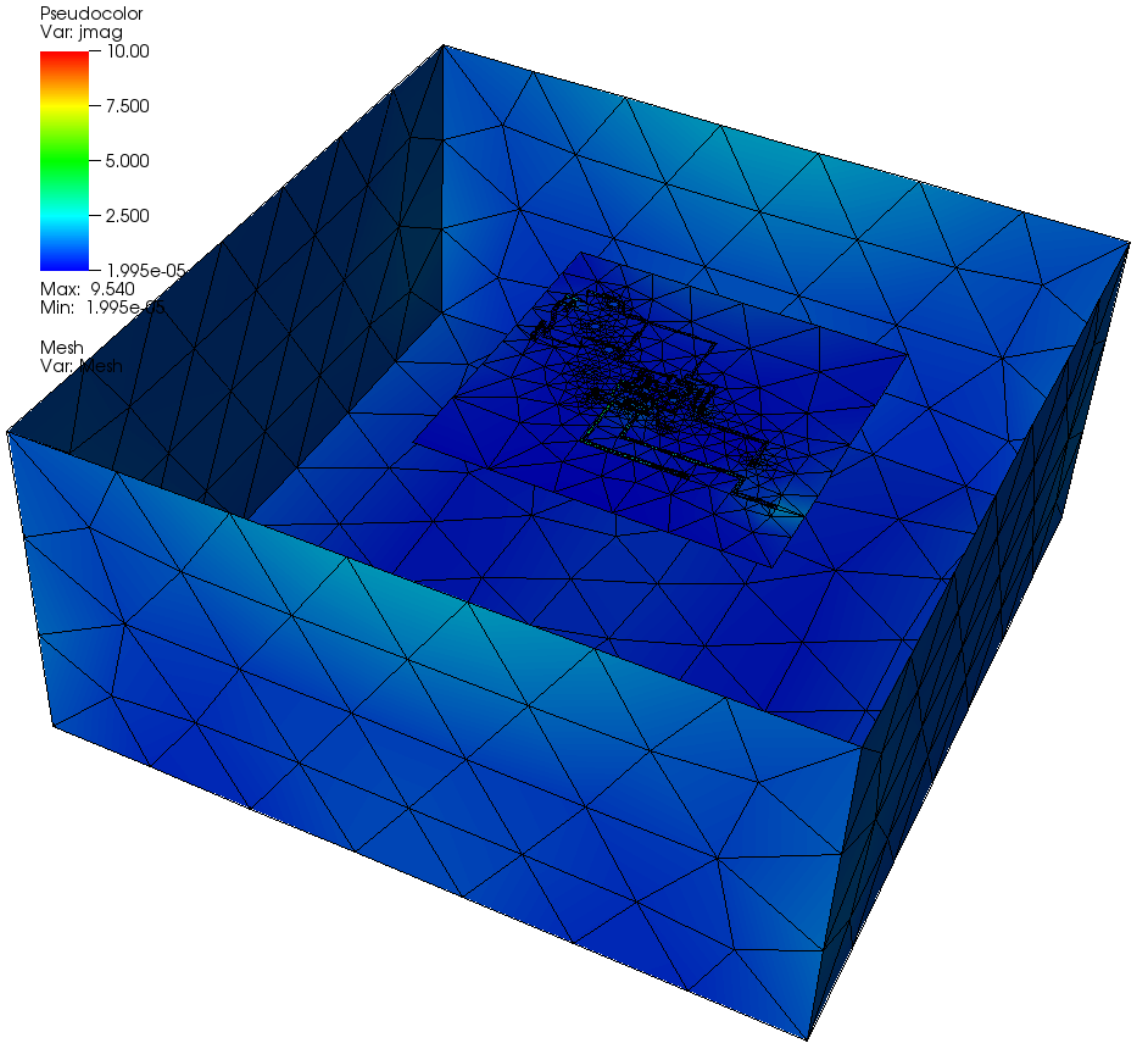

Figure 9. Computed electric currents. The top of the box has been removed from the graphic to show the currents on the circuit.

The first experiment is the aperture location given by X and Y coordinates, thus there are only two parameters. For convenience the parameters are scaled to [0,1], with the transformation X = 60 * p1 - 30, Y = 60 * p2 - 30. To get started the 2-dimensional parameter space was partitioned into 4 quadrants, and each ROM was constrained to use no more than 15 sample points. The total number of sample points was 43, thus 43 full complexity solutions were required to build the ROM. The quality of the ROM was tested by computing the truth solution at 200 random points, the mean error is 0.02, the worst case error is 0.19. This may or may not be adequate depending upon the application. In order to demonstrate the adaptive refinement of parameter space, the tolerance was reduced to 0.001. The result is shown in Figure 10 and 11 below, the parameter space was subdivided into 16 partitions. The maximum ROM dimension

was again constrained to be 15, and a total of 154 full complexity solutions were required to build the ROM. This second model was tested using 200 random truth solutions, the median error was 0.0025 and the worst case error was 0.06. The cumulative error distribution is shown in Figure 12. Comparing the two models, the second is clearly more accurate than the first, at the same on-line cost, but at a larger off-line cost. Note that the cost of evaluating the ROM is 10^6 times less expensive than the full complexity solution, it is essentially free. The ROM can be evaluated millions of times, essentially completely exploring all of the 2D parameter space, at a one time cost of 154 full complexity solves to build the ROM. And note that even though the reduced order model is of dimension 15, it provides the full 4122 unknowns. Other than some "noise", the ROM solution is indistinguishable from the full solution.

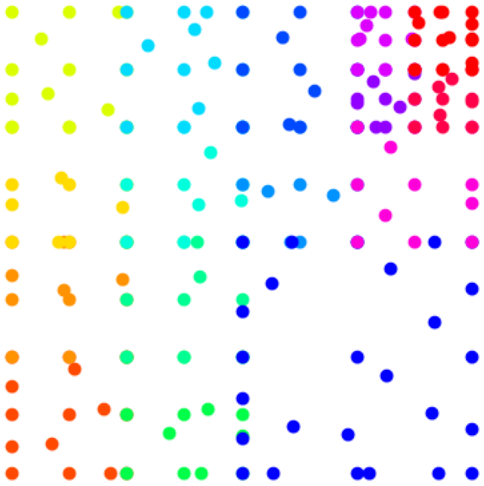

Figure 10. Sample points for the aperture location.

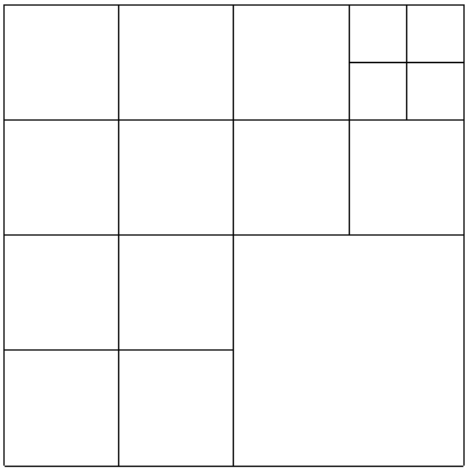

Figure 11. Partitioning of the parameter space. There is one small dimensional model for each partition. The overall model is then a collection of small models.

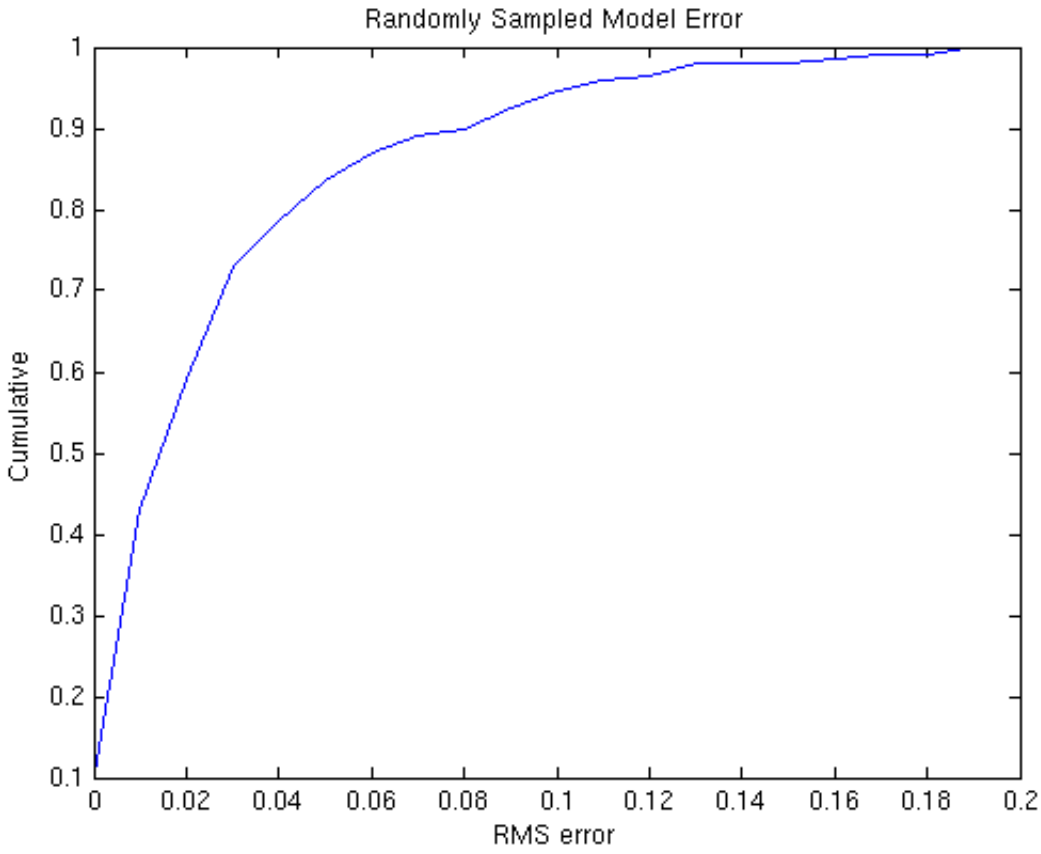

Figure 12. Error in the reduced order model, computed by differencing the ROM with the truth solution for 200 random points in parameter space.

A second test was performed, this time using the X, Y, and Z location of the circuit in the box as parameters. The parameter transformation is X = 60* p1, Y = 60* p2, Z = 20*p3 + 10. The ROM tolerance was set to 0.001 and adaptive hierarchical refinement of parameter space was employed. The maximum number of sample points per partition was limited to 35. The resulting partitioning of parameter space is shown in Figure 13, with 43 partitions. The sample points are shown in Figure 14. The total number of points is 605. Using these values, the ROM is 3e5 times faster than the full complexity solution. The error of the ROM was evaluated by computing the truth solution at 200 random points, the median error is 0.01 and the worst case error is 0.055. A histogram of the error is shown in Figure 15. If the ROM is evaluated for points on a 20x20x20 grid in parameter space, it would be 47 times more efficient than the brute force approach. If the ROM is evaluated on a 100x100x100 grid in parameter space, it would be 950 times more efficient than the brute force approach.

The above results are very promising. But there were some issues. One issue is that the error estimator that we used was not particularly accurate, and it was optimistic (the actual error was always greater than the estimated error). Also, it was very difficult to achieve a highly accurate ROM, we were unable to achieve a worst-case error below 0.01. We suspect this is due to the nature of the BEM method, the full complexity equations are ill conditioned, hence small errors are amplified. There are better conditioned numerical methods for frequency domain electromagnetics than those used in the LLNL EIGER code, and perhaps this would enable more accurate ROM's. But this was beyond the scope of this project.

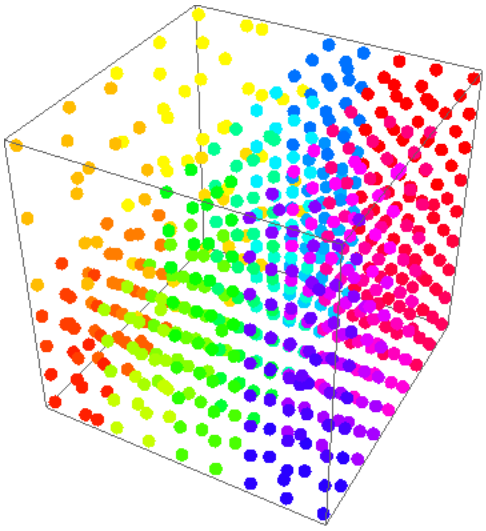

Figure 13. Sample points in parameter space for the (X,Y,Z) circuit location.

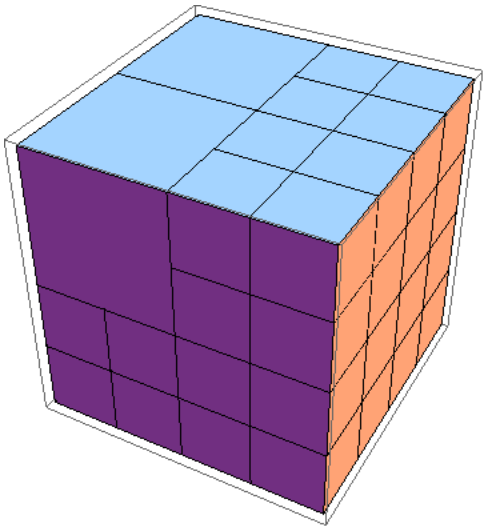

Figure 14. Hierarchical partitioning of the 3D parameter space. There is one model per partition.

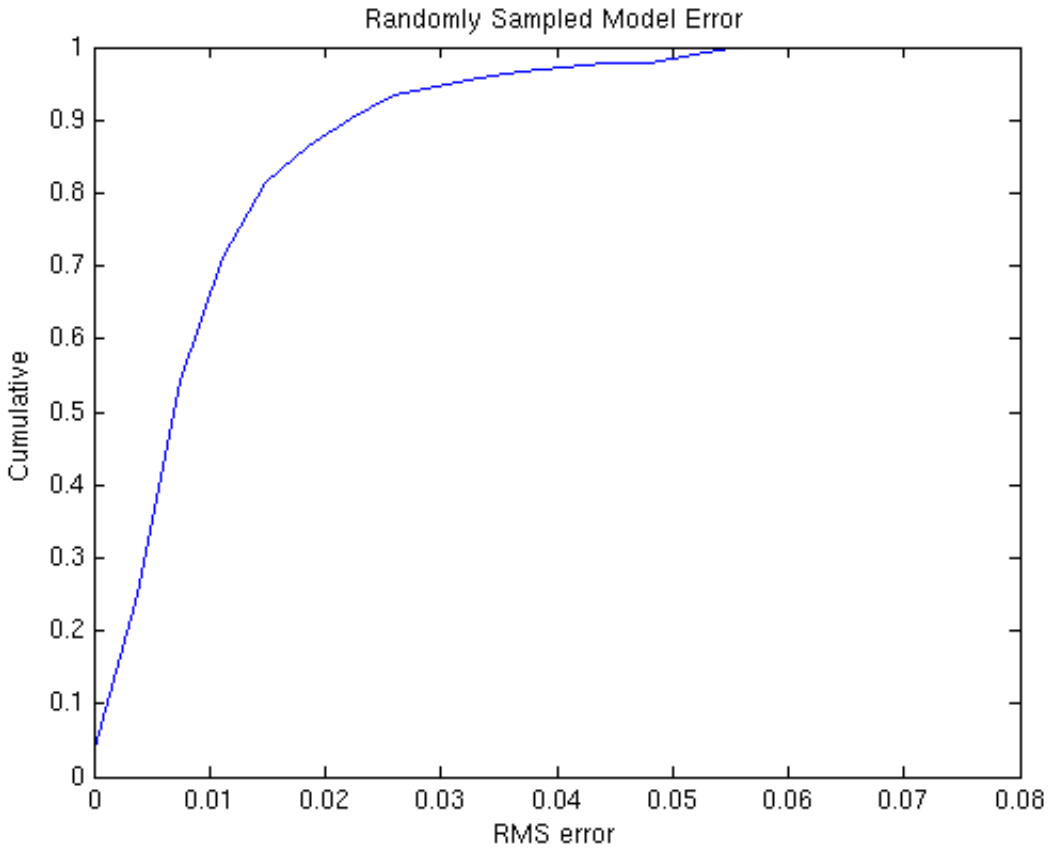

Figure 15. Error in the ROM computed at 200 random points in parameter space.

4. Photonic Crystal Fiber Results

Photonic Crystal Fibers (PCF), also known as microstructural fibers or holey fibers, have evolved from an academic curiosity into commercial technology, with wide ranging applications in telecommunications, biotechnology, metrology, etc. The key aspect of PCF's is arrays of small holes in the otherwise homogenous glass. The size, shape, and location of the holes are parameters that can in principle be optimized for a particular application. In addition to optimization, it may be important to understand the effect of geometric perturbations from the ideal, due to manufacturing issues, bending of the fiber, etc. There are basically two distinct types of PCF's, solid core and air core. Solid core fiber consists of a solid glass cylindrical fiber, with a cladding that consists of the same glass material but with an array of longitudinal holes surrounding the solid core. The holes essentially reduce the effective permittivity of the glass, thereby enabling total internal reflection guiding of light in the core much like standard fibers or graded-index optical fibers. In air core fibers, also called Photonic Bandgap Fibers, there is a large hole in the center of the fiber. The surrounding cladding still consists of glass with an array of holes, but the parameters are such that the cladding has an negative effective permittivity at the desired frequency, resulting in a bandgap and prohibition of wave propagation in the

cladding. The result is that the light is trapped in the air core. PCF's can be designed to have very broadband single mode operation, zero or anomalous dispersion relations, a wide range in numerical apertures, and spatially broad fields resulting in high power at low power densities.

The appropriate equation for simulating an optical fiber is Maxwell's Equations, with the assumption that the solution is a wave propagating in the z-direction. Thus only a 2D mesh is required, this gives the solution of the electric field in the plane normal to the direction of propagation. The problem is a generalized eigenvalue problem involving a mass-like matrix and a stiffness-like matrix, the eigenmode is the electric field and the eigenvalue is the wavenumber (actually the eigenvalue is the wavenumber squared). We are concerned with the largest eigenvalues and associated eigenmodes, these are the smoothest modes, and the eigenvalues are purely real. There exists higher order modes, and complex modes which do not propagate, and these are of no interest here.

$$k^2 \frac{1}{\epsilon} \vec{D}_T = \nabla_T \left(\frac{1}{\epsilon} \nabla_T \cdot \vec{D}_T \right) - \nabla_T \left(\nabla_T \times \frac{1}{\epsilon} \vec{D}_T \right) + \omega^2 \mu \vec{D}_T$$ Transverse component of E-field

$$\begin{aligned} H_z &= \frac{1}{i\omega\mu} \nabla_T \times \frac{1}{\epsilon} \vec{D}_T && \text{Longitudinal component of H-field} \\ \vec{H}_T &= \frac{1}{ik} \nabla_T H_z + \frac{\omega}{k} \hat{z} \times \vec{D}_T && \text{Transverse component of H-field} \\ D_z &= -\frac{1}{i\omega} \nabla_T \times \vec{H}_T && \text{Longitudinal component of E-field} \end{aligned}$$

$$\vec{D}_T = \sum_i^n x_i \vec{W}_1^i$$ Finite element approximation

Generalized Eigenvalue Problem

$$Ax = k^2 Bx \quad \text{where} \quad \begin{cases} A &= \omega^2 M_T(\mu) + S_T\left(\epsilon^{-1}\right) \\ & \quad -M_1(\mu) T_{01} M_0^{-1}(\mu) T_{01}^T M_T\left(\epsilon^{-1}\right) \\ B &= M_T(\epsilon^{-1}) \end{cases}$$

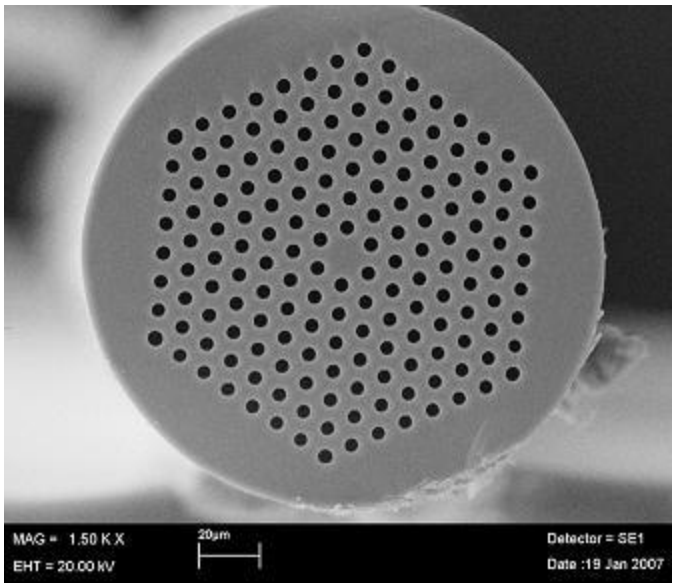

Figure 16. An example photonic crystal fiber.

Figure 17. The LLNL fiber draw tower.

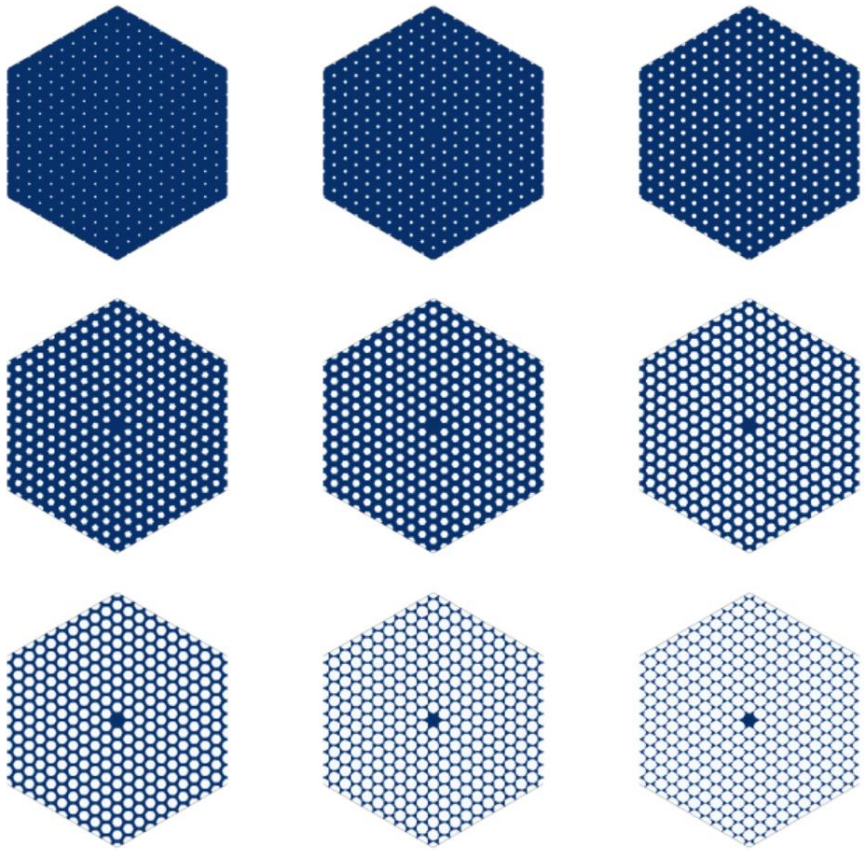

Figure 18. Varying the hole radius.

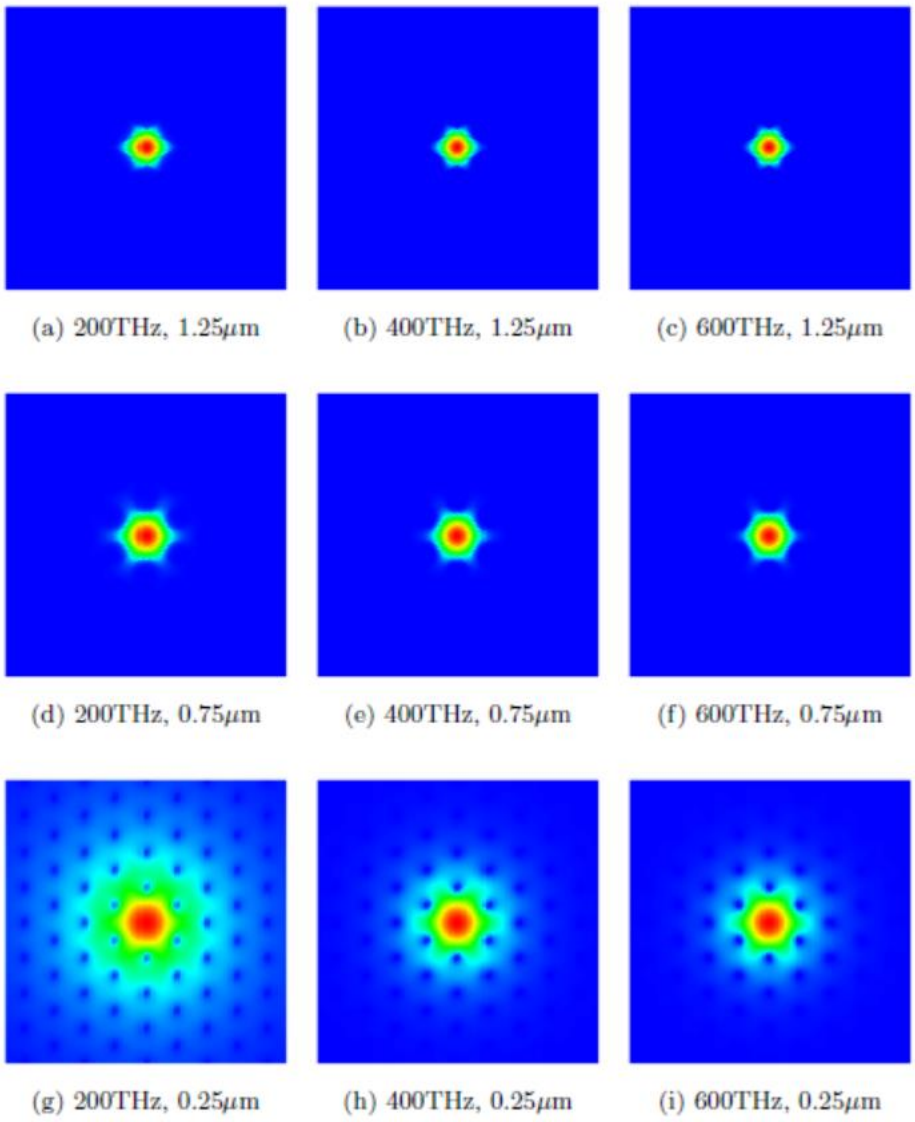

Figure 19. Computed eigenmodes (electric field intensity) for various frequencies and hole radii.

To get started we assume the parameters are the frequency which will range from 200 to 600 THz, and the hole radius which will vary from 0.25 to 1.25 microns. The center to center distance of the holes will remain fixed at 2.5 microns. Other hole separations can be considered by simply scaling these results. The permittivity of the glass will be 2.25. The discrete problem has 360,000 unknowns. Thus the dimension of the eigenvalue problem is N=360,000. ARPACK is used to compute the two largest eigenvalues using a Lancoz space of 20, each solve takes about 5900 seconds (1 1/2 hour) on a multiprocessor computer (12 processors I think).

The eigenmodes vary in an interesting manner, as the hole radius is increased the field energy is concentrated in the center core. This is because the effective permittivity is reduced in the outer core. The same thing happens as the frequency increases, this is shown in Figure 19

The algorithm for constructing the reduced order model for this fiber problem is similar to that used in the circuit problem, we use a change of basis plus radial basis function interpolation. One improvement made here is that the sample points in parameter space lie on a nested grid. We start with a 2x2 grid, then move to 3x3, 5x5, 9x9, and 17x17. Not all points are used, an error estimator is used to determine when to use a new sample point. This nested grid approach is used because for each level as different scale factor is used for the radial basis functions. Figure 20 shows the sample points in 2D parameter space. The larger disks represent the early sample points, the smaller points represent the latter sample points. All sample points are used to construct the ROM. Using 57 sample points, the worst case error is less than 1%. Error plots for various levels of refinement are shown in Figure 21. Note that the ROM requires only a few seconds to evaluate.

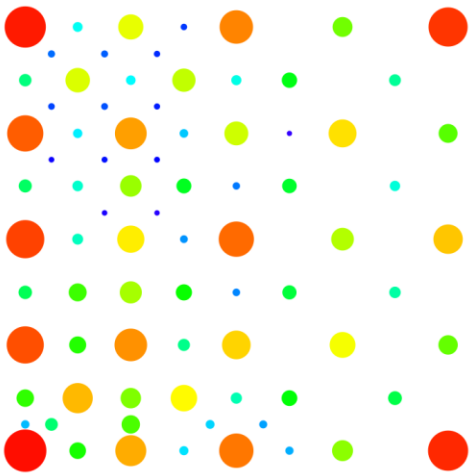

Figure 20. Sample points in 2D parameter space. Large dots denote the early samples, small dots denote the latter samples. Note the non-uniform sampling.

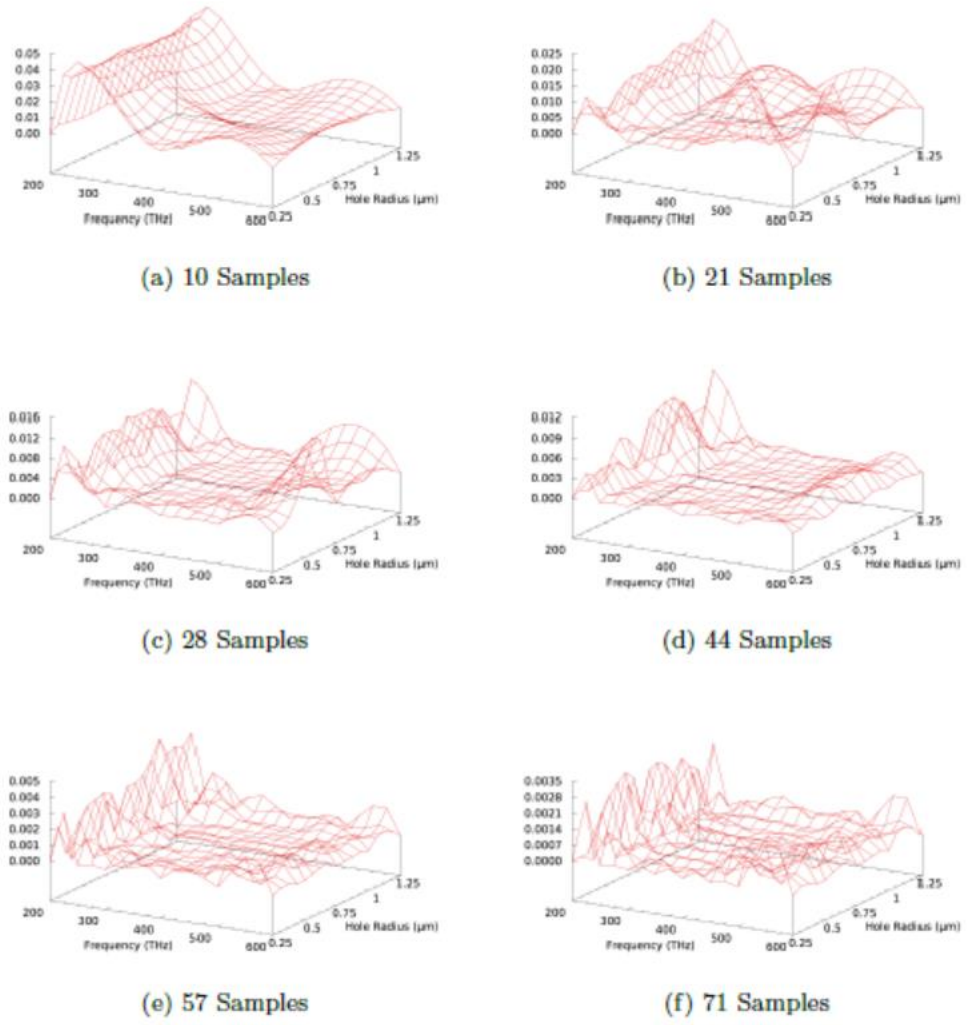

Figure 21. Error in the reduced order model for different size models.

We have preliminary results for a three parameter study, frequency, hole radius, and stretch factor. The ROM required 260 snapshots, and the dimension of the reduced order model is M=472, tiny compared to the original dimension of N=360,000. The ROM is about 844 times faster, and the error is less that 0.24% Some results are shown below. In summary, these results are very impressive, in part because the parameter space sampling is a bit more sophisticated that that used for the circuit problem, and also because it seems that the finite element electromagnetic eigenvalue problem is better conditioned.

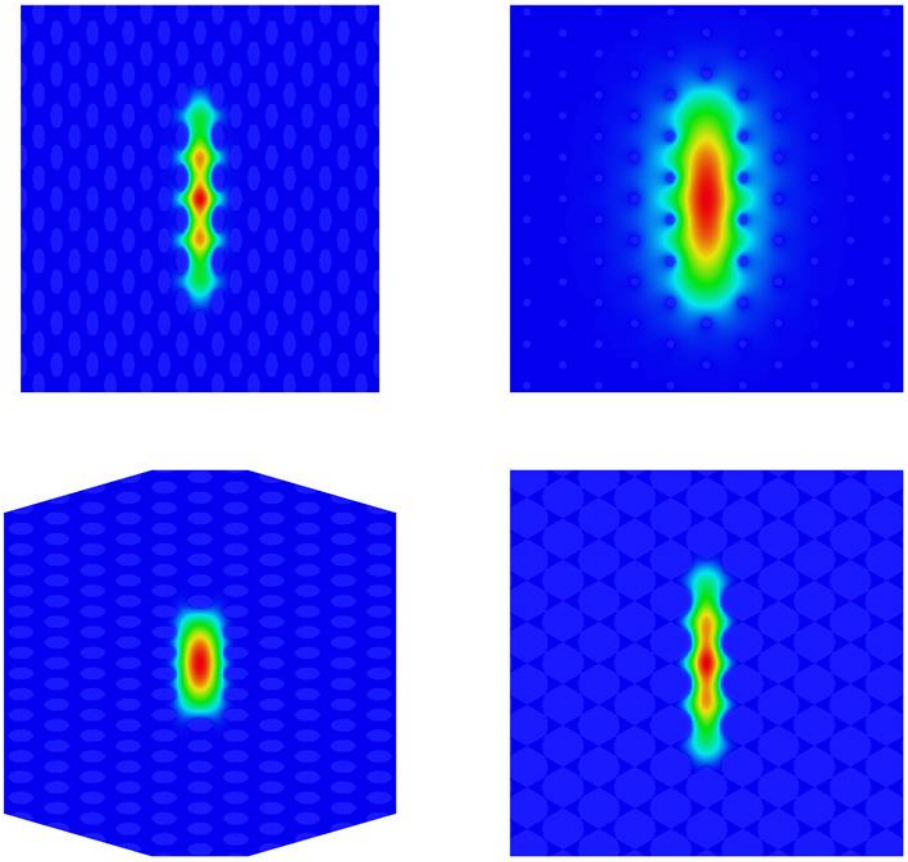

Figure 22. Example eigenmodes (electric field intensity) for various hole radii and stretch factors.

5. Microstrip Waveguide Results

This problem is a microstrip waveguide design problem parameterized by three geometric parameters. Depending upon the value of these parameters, the waveguide can act as a through, a lowpass filter, or a bandpass filter. The problem will be solved using the frequency domian boundary element code EIGER. Only the metal microstrip is meshed, the dielectric substrate is modeled using a Greens function. The computational mesh consists of 720 triangles, and 1290 unknowns, quite small. But for this problem, due to the Greens function treatment of the substrate, the matrix fill time completely dominates. A single solve requires several minutes.

For this problem the empirical interpolation method will be used instead of radial basis function interpolation. The three dimensional parameter space will be sampled on a 5x5x5 uniform grid (probably not optimal, this was just a starting point). This resulted in 125 snapshots from which to construct the truncated proper orthogonal decomposition. The SVD of these snapshots is

shown in Figure 23. The error tolerance was set to 10^-6, resulting in M = 71 for the change of basis. The dimension of the ROM will be M = 71, 5% smaller than the dimension of the original full complexity problem. For an O(n^3) algorithm the speedup is then around 6000x, but for this problem the matrix fill dominates so the speedup is even greater. These same sample points were used for the empirical interpolation. For each snapshot, the impedance matrix is reshaped into a vector and stored as a column in a new matrix V. The SVD of this matrix is then computed. The empirical interpolation had a tolerance of 10^-6, and resulted in 35 magic points. The SVD is shown in Figure 24, and the magic points are shown in Figure 25. Recall that the EIM process results in basis function expansion of the parameterized impedance matrix, where the basis are impedances matrices for specially selected parameters. The complete reduced order model has the form

$$Q(p) = a1(p)\, Q1 + a1(p)\, Q2 + \ldots a35(p)\, Q35$$

where $Qk = V^T Zk V$. Recall that V is the change of basis of dimension m = 71, hence the Q matrices are 71 x 71. The coefficients ak(p) are the values of the impedance matrix Z(p) evaluated at magic point k.

An example truth solution, and an example ROM solution, are shown in Figures 26 and 27, respectively. They are indistinguishable. For computing the error the truth solution was computed on a finer grid, and the error is shown in Figures 28 and 29 for the two sources of error (the error in the change of bases and the error in the empirical interpolation can be computed separately). About 90% of the fine grid points have an error of less that 10%, not bad, but not great either. The error plots indicate that the empirical interpolation the dominate source of the error. We suspect the problem with the error is similar to that of the circuit problem, ill conditioning of the original full complexity discretization. It would be interesting to do an apples-to-apples comparison of radial basis function verses empirical interpolation for this problem.

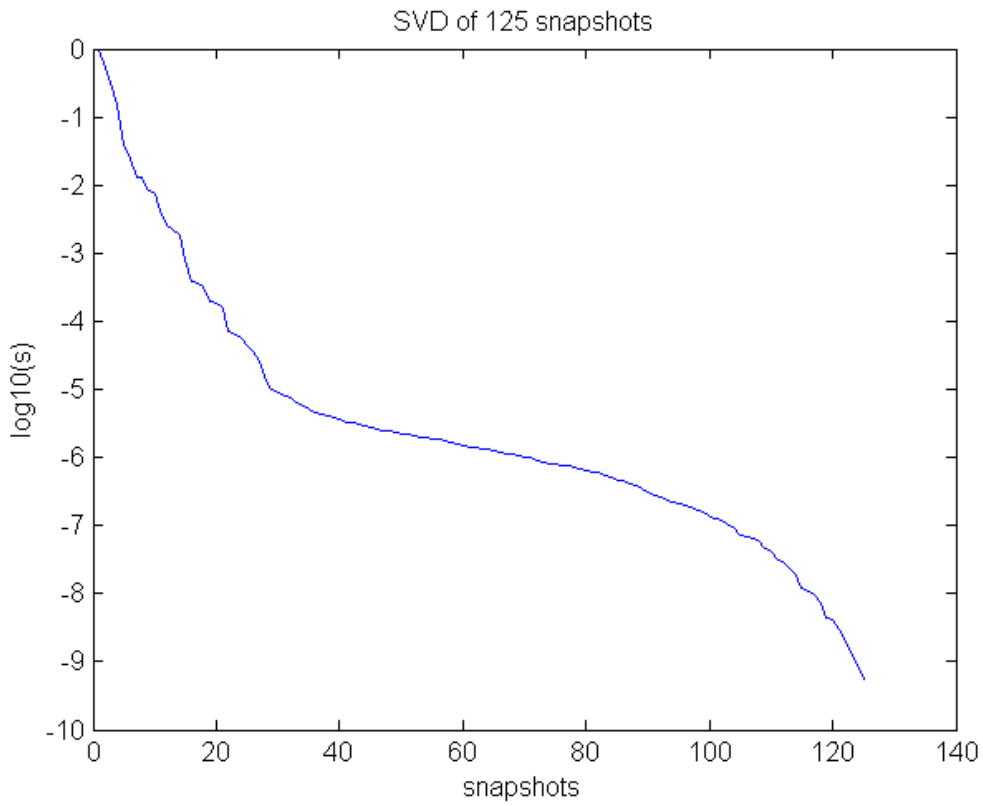

Figure 23. Singular value decomposition of solution snapshots for 125 sample points.

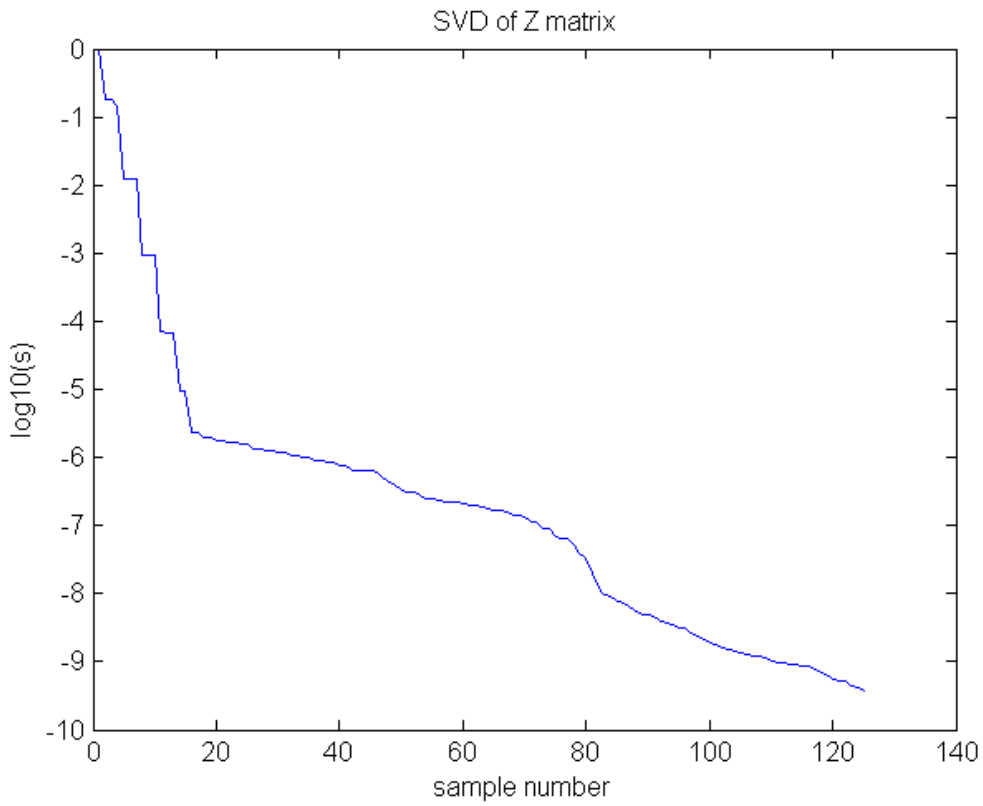

Figure 24. Singular value decomposition of the impedance matrix for 125 different snapshots.

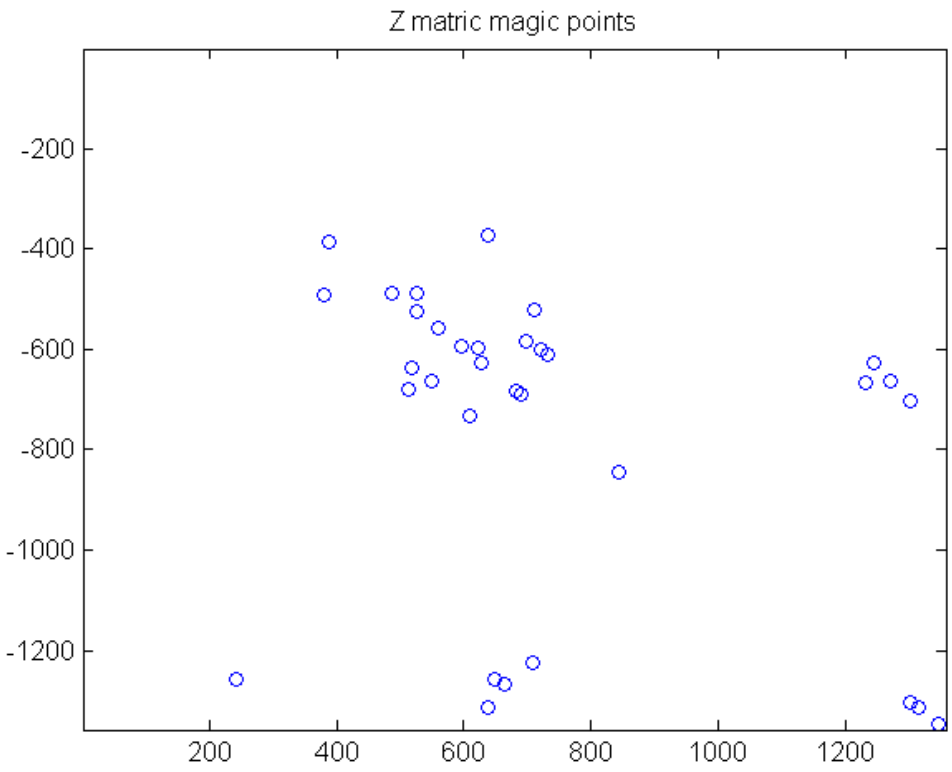

Figure 25. Computed magic points for sampling the impedance matrix.

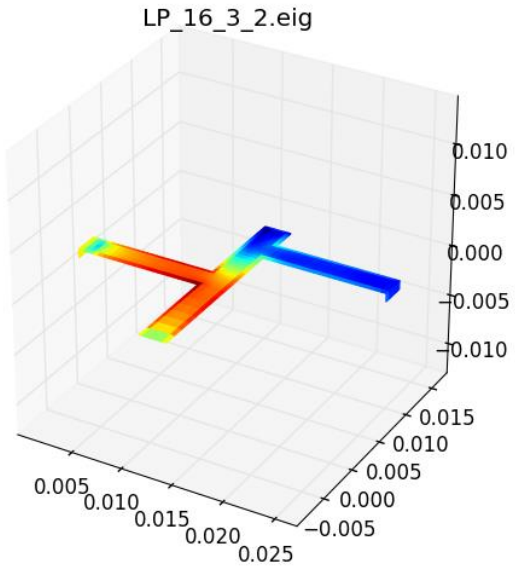

Figure 26. Truth current density for one point in parameter space.

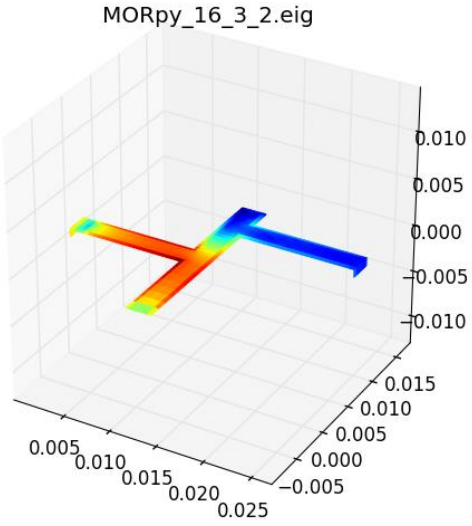

Figure 27. Model current density for one point in parameter space.

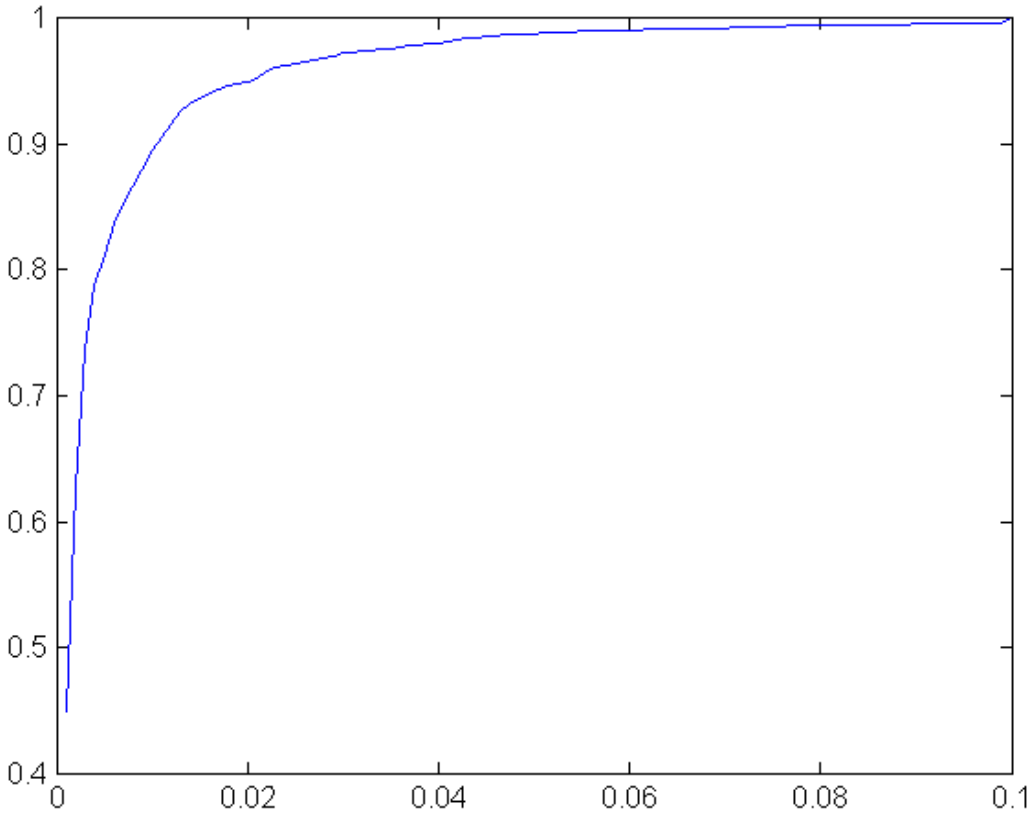

Figure 28. The error in the change of basis.

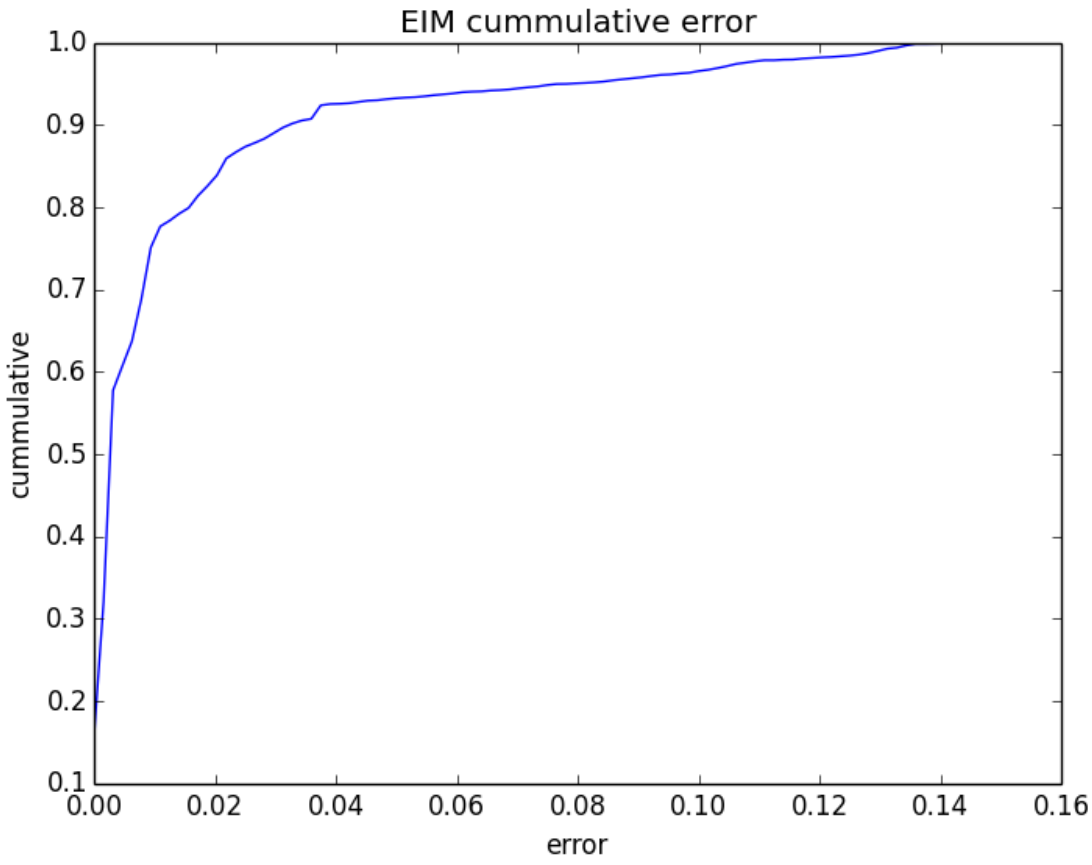

Figure 29. The error in the complete reduced order model.

6. Mechanical Microstructure Results

This experiment has nothing to do with electromagnetic interactions. Instead, we wanted to see if the methodology developed for electromagnetic interactions would work for structural analysis problems. The structure of interest is a unit cell of a negative coefficient of thermal expansion material. The structure consists of two distinct materials, each with a standard positive coefficient of thermal expansion. The structure is hinged in such a way that it the outer boundary contracts as the beam elements expand. This structure was simulated using ALE3D. The ALE3D code was modified to write the stiffness matrix to disk for analysis. The structure is shown in Figures 30 and 31 for two different points in parameter space. There are several different parameters that can be varied, in this study we only varied 2 parameters. The material is considered to be “at rest” at 270 degrees K, and a small temperature change (10 degrees) is applied to the entire problem. This results in a small displacement, and example displacement is shown in Figure 32.

The structural problem is discretized using a 22,000 element hexahedral mesh. The resulting sparse stiffness matrix had about 690,000 nonzero entries. The parameter space was sampled on a 22x22 grid, solutions on this fine grid will be considered the truth solutions. Empirical interpolation will be used to approximate the stiffness, using snapshots from a 3x3 grid, a 5x5 grid, and an 11x11 grid. The results are shown in Figure 33 - 35 below. The error is 1.4% for a 3x3 grid (model of dimension m = 9), 0.07% for a 5x5 grid (model of dimension m = 25), and 0.004% for a 11x11 grid (model of dimension m = 121). These are very small errors, thus we see that empirical interpolations works very well for the FEM discretization of linear elasticity. Of course the total error is related not just to the empirical interpolation error, but also the conditioning of the FEM matrix. Time ran out before the total error could be computed.

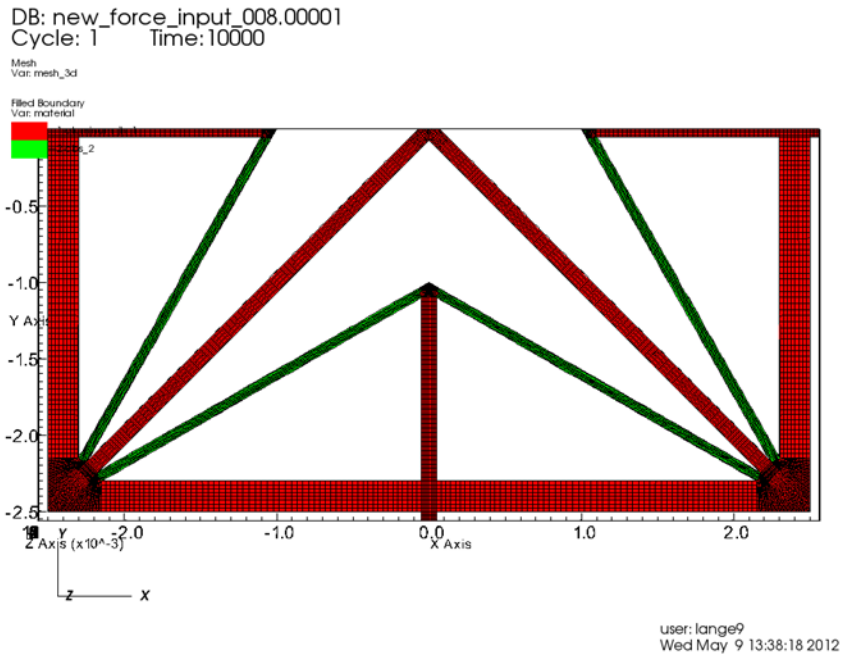

Figure 30. Microstructure for one set of parameters. The green and red denote the two different materials.

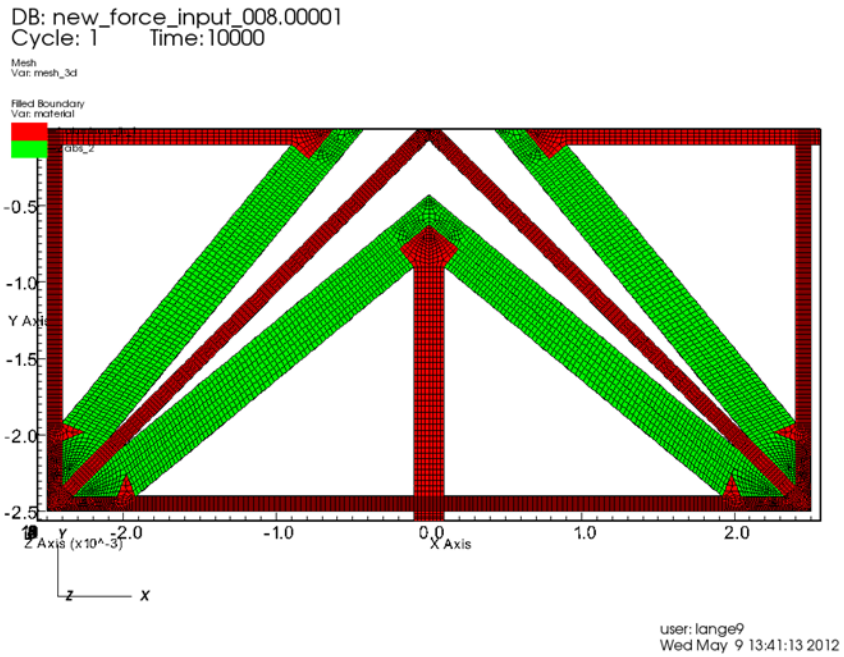

Figure 31. Microstructure for a different set of parameters.

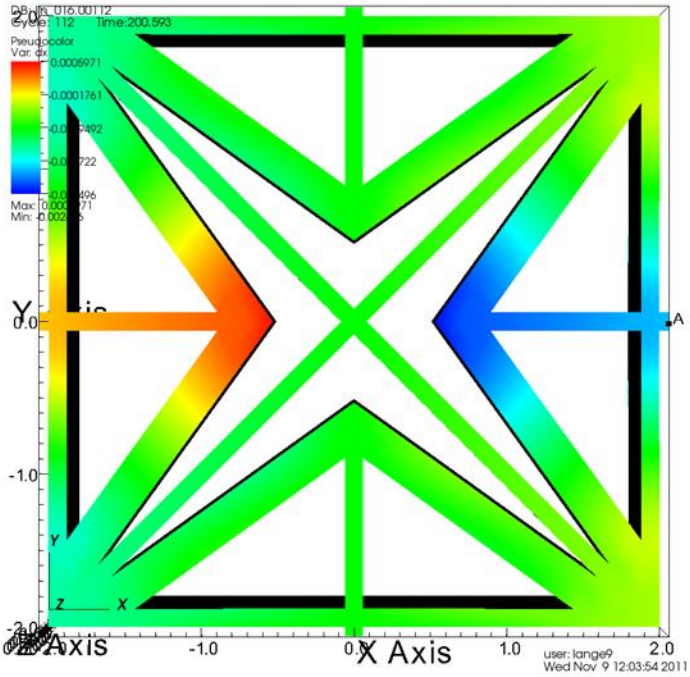

Figure 32. Computed displacement (x component) for a 10 degree temperature change. The net displacement it determined by measuring the displacement at two tabs on the y=0 line.

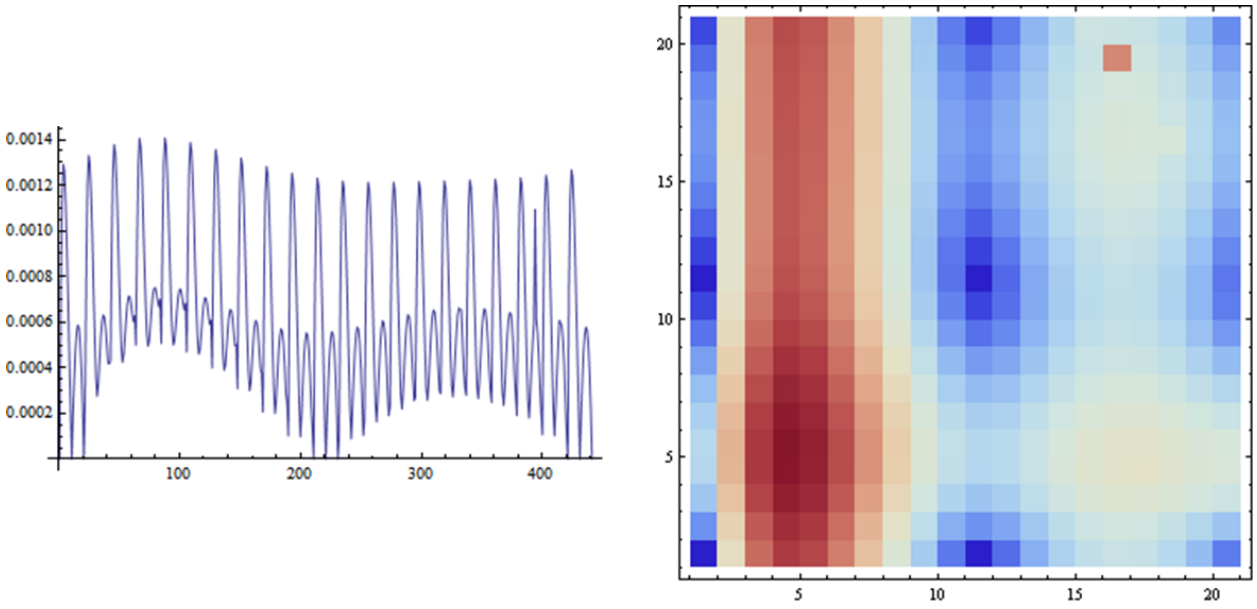

Figure 33. Empirical interpolation error for a 3x3 grid of sample points.

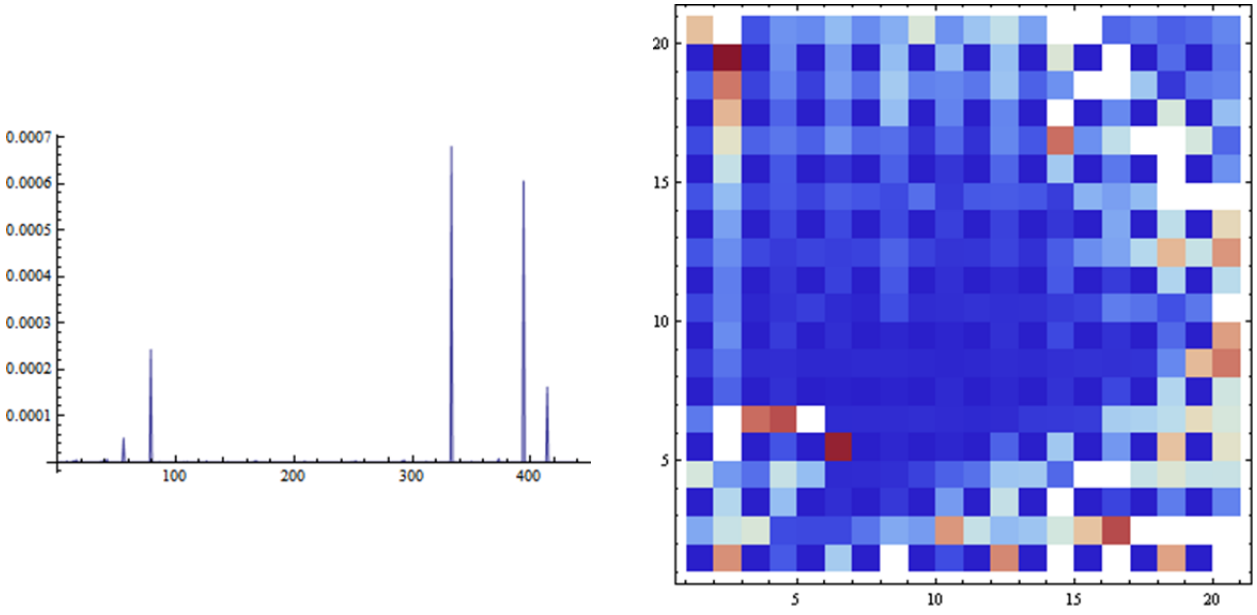

Figure 34. Empirical interpolation error for a 5x5 grid of sample points.

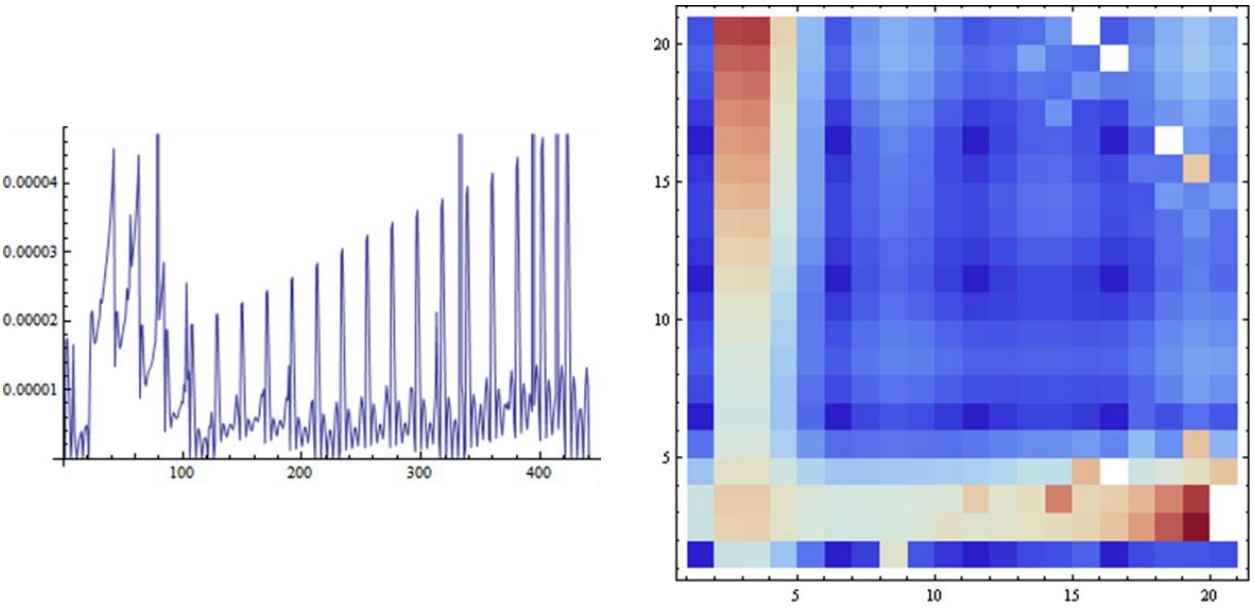

Figure 35. Empirical interpolation for a 11x11 grid of sample points.

7. Future Directions

We demonstrated that model order reduction can be very useful for exploring parameter space. Excellent results were obtained for an electromagnetic eigenvalue problem, a decrease in run rtime of over 800x, with less than 1% error. Barely satisfactory results were obtained for other electromagnetic interaction problems, we suspect this is more an issue of ill-conditioning of the original full complexity solution, rather than a flaw in the model order reduction algorithm.

Clearly one possible improvement is a more clever sampling of parameter space. If good error estimators can be developed, these can be used to guide the choice of sample points. Alternatively a Latin hypercube or related sampling can be employed. Note that there is nothing statistical about our problem or process, but sample points that are optimal for statistical sampling may work well in any case.

The computational experiments we performed were limited in size in order for the truth solution to be tractable. The real benefit in model order reduction is for large parameter spaces. Our algorithms are independent of the dimension of the parameter space, and testing these algorithms for larger problems should be a priority.